Machine learning-based approaches for tomato pest classification

Gayatri Pattnaik[1], Kodimala Parvathi[2]
[1]School of Electronics Engineering, KIIT Deemed to be University, Bhubaneswar, India.
[2]Department of Electronics and Communication Engineering, Chaitanya Engineering College, Visakhapatnam, India

Article Info

Article history:

Received Jan 22, 2021
Revised Jan 31, 2022
Accepted Feb 08, 2022

Keywords:

DT
GLCM
HOG
K-NN
LBP
SURF
SVM

ABSTRACT

Insect pests are posing a significant threat to agricultural production. They live in different places like fruits, vegetables, flowers, and grains. It impacts plant growth and causes damage to crop yields. We presented an automatic detection and classification of tomato pests using image processing with machine learning-based approaches. In our work, we considered texture features of pest images extracted by feature extraction algorithms like gray level co-occurrence matrix (GLCM), local binary pattern (LBP), histogram of oriented gradient (HOG), and speeded up robust features (SURF). The three standard classification methods, including support vector machine (SVM), k-nearest neighbour (k-NN), and decision tree (DT) are used for classification operation. The three classifiers have undergone a comprehensive analysis to present which classifier with which feature yields the best accuracy. The experiment results showed that the SVM classifier's precision using the feature extracted by local binary patterns (LBP) algorithm achieves the highest value of 81.02%. MATLAB software used for feature extraction and waikato environment for knowledge analysis (WEKA) graphical user interface for classification.

Corresponding Author:

Gayatri Pattnaik
School of Electronics Engineering, KIIT Deemed to be University
Bhubaneswar, Odisha, India, Pin-751024
Email: gayatripattnaik13@gmail.com

1. INTRODUCTION

Insect pests cause crop losses every year which cost around US$36 billion. Therefore, immediate decisions are required to prevent pest proliferation. Traditionally, the pesticide was used to avoid damage to the crop, but excess use of it is hazardous and detrimental to our ecosystem. Agricultural scientists created a scheme called integrated pest management (IPM) to limit the use of chemical pesticides since the 1960s [1]. Although it has effective and accurate ways of pest control, still it is not compatible all the time. It needs thorough observation of pest behaviours. Therefore, automatic detection and classification of pest images based on image processing are proposed in this study to achieve pest identification and control.

In recent years, machine learning (ML) and image processing methods have been explored for automatic pest detection and classification mechanism. Wang *et al.* [2] implemented two ML algorithms artificial neural network (ANN) and support vector machine (SVM) to learn pest features and obtained good results. Fina *et al.* [3] developed a k-mean clustering and correspondence filter-based automated plant pest identification and recognition technique. Furthermore, Xie *et al.* [4] developed an insect recognition system using advanced multiple task sparse representation and multiple kernel learning techniques to classify 24 crop pest types. He *et al.* [5] demonstrated a classification approach based on machine vision and image processing for identifying cotton pests and diseases. It uses picture enhancement and filtering algorithms

to calculate the damage ratio of cotton leaves caused by pests. Dey *et al.* [6] evaluated the use of different ML models namely SVM, ANN, Bayesian classifier, binary decision tree (BDT) and k-nearest neighbor (k-NN) for classification of a pest as whitefly which affects on the various plant. Another harmful pest as thrips in the strawberry plant was detected using the SVM classification method by Ebrahimi *et al.* [7] and reported a mean percent error (MPE) parameter of less than 2.25%. Xiao *et al.* [8] put forward a classification method to identify four vegetable pests using a bag of words and SVM techniques. By using this method, they obtained an average accuracy of 91.56%. In the last few years, a machine learning-based deep learning convolutional neural network (CNN) was utilised to categorise 82 types of common pests with an accuracy of 91% [9].

We thus propose an automatic system based on image processing and ML for pest detection and classification of a single crop as a tomato crop. India occupies the second position in the area and the production of tomatoes. Yearly tomato production in India is amounted to over 20 million metric tonnes [10]. Tomato crop suffers yield losses of 10.8%, which costs about US$36 billion. Besides, less research in the field of tomato pest detection and classification has been proposed. So, all our efforts have been made for the minimization of losses due to tomato pest. Tomatoes affected by borer insect were identified by image processing methods like segmentation and morphological operation [11]. In addition, a pest named Tuta absoluta [12] was detected by symptoms that are caused due to viruses in the tomato plant. Another harmful insect pest in tomato greenhouses is the whitefly (Bemisia tabacii). They were detected and classified by machine learning shallow models like k-NN and multilayer perceptron (MLP). The MLP classifier achieved the highest accuracy of about 81.12% [13]. A system has been proposed [14] for the tomato fruit grading system. The design system included three phases: pre-processing, feature extraction by gray level co-occurrence matrix (GLCM), and SVM classification. Rupanagudi *et al.* [15] demonstrated another method of detecting borer insects on tomato plants using cloud computing-based frameworks. Besides some insect pests, a particular or combination of viruses is also having some binding critical effect on tomato plant growing as in [16]. This paper presents an automatic system based on image processing and ML for pest detection and classification of a single crop as a tomato crop with limited data.

2. MATERIALS AND METHODS

2.1. Materials

Images of tomato pests were gathered from a variety of web sources, including flickr, insect images, IPM image, National Bureau of Agricultural Insect Resources (NBAIR), and Tamil Nadu Agricultural University (TNAU). Figure 1(a) Bactrocera latifrons, Figure 1(b) Bemisia tabacii, Figure 1(c) Chrysodeixis chalcites, Figure 1(d) Epilachna vigintioctopunctata, Figure 1(e) Helicoverpa armigera, Figure 1(f) Icerya aegyptiaca, Figure 1(g) Liriomyza trifolii, Figure 1(h) Nesidiocoris tenuis, Figure 1(i) Spodoptera litura, and Figure 1(j) Tuta absoluta are the 10 insect pests that mostly harm tomato plants. Sample images from each class have been presented in Figure 1. Dataset considered in this study includes 859 images of tomato pest. The number of images for each class has been provided in Table 1 as details of the dataset.

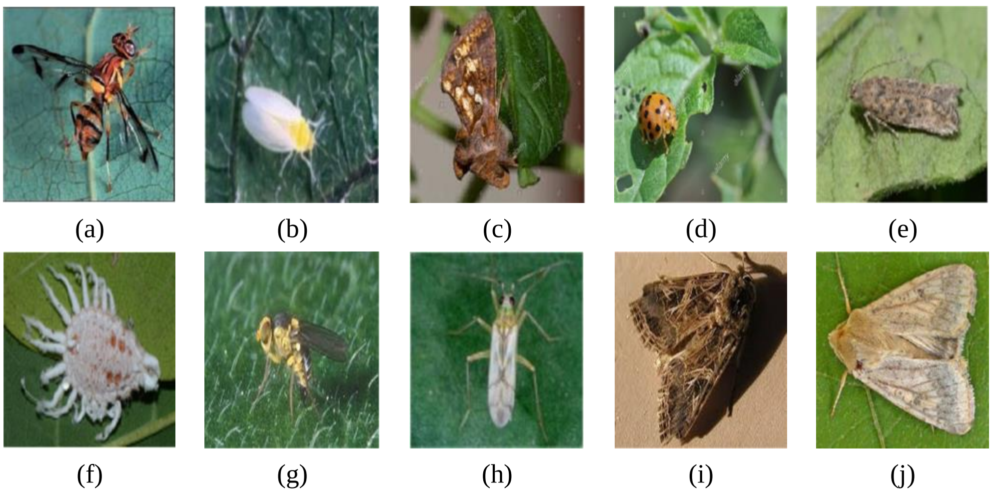

Figure 1. Sample image from each class of tomato pest: (a) Bactrocera latifrons, (b) Bemisia tabacii, (c) Chrysodeixis chalcites, (d) Epilachna vigintioctopunctata, (e) Helicoverpa armigera, (f) Icerya aegyptiaca, (g) Liriomyza trifolii, (h) Nesidiocoris tenuis, (i) Spodoptera litura, and (j) Tuta absoluta

Table 1. Details of tomato pest

Class label	Class name	#Images
Pest 1	Bactrocera latifrons	80
Pest 2	Bemisia tabacii	80
Pest 3	Chrysodeixis chalcites	94
Pest 4	Epilachna vigintioctopunctata	94
Pest 5	Helicoverpa armigera	92
Pest 6	Icerya aegyptiaca	80
Pest 7	Liriomyza trifolii	88
Pest 8	Nesidiocoris tenuis	91
Pest 9	Spodoptera litura	80
Pest 10	Tuta absoluta	80
Total number of images		859

2.2. Methods

This section presents the methodology used for tomato pest image classification. It consists of five major parts: i) image acquisition, ii) image pre-processing, iii) image segmentation, iv) feature extraction, and v) classification. The basic procedure of our approach is represented in Figure 2.

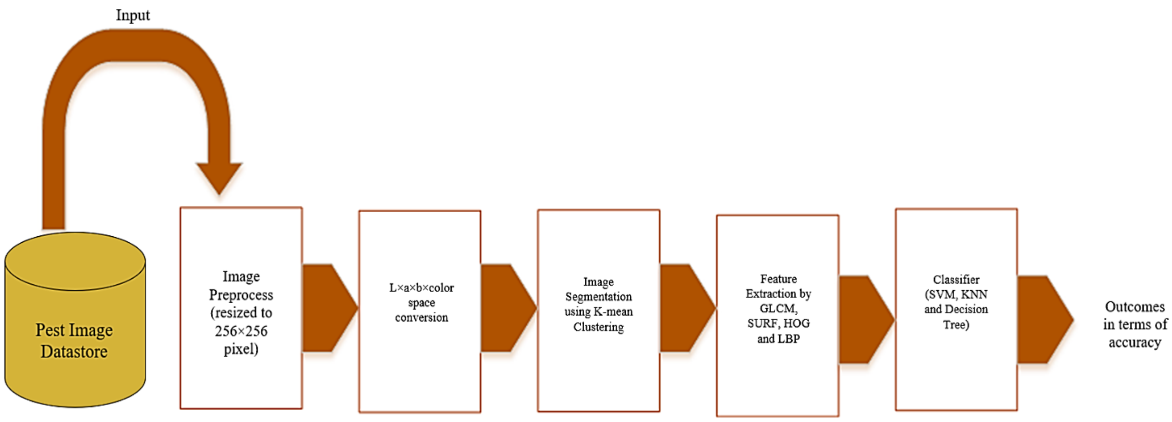

Figure 2. Block diagram of the proposed approach

2.2.1. Image acquisition

Images of tomato pests are gathered from the internet and saved in a database. A total of 859 image samples about 10 different common types of tomato pest of India were collected from the internet. These are used to train the ML classifier to automatically identify the pest.

2.2.2. Image preprocessing

Then images are applied for pre-processing to modify image data. Pest images are obtained in red, green and blue (RGB) format. Here we used to follow two steps: i) resizing RGB image to 256×256 pixels of standard size, and ii) conversion from RGB to L×a×b color space image transformation.

2.2.3. Image segmentation

Segmentation is the task of partitioning an exciting part from the background. In our approach k-mean clustering algorithm was used, to segment the intended area like pest from location consisting of a stem, and leaves. The k-mean clustering algorithm tries to classify objects into k number of classes according to a set of features.

2.2.4. Feature extraction

Feature extraction plays a vital role in obtaining the desired object. Our proposed work adopted feature extraction techniques like GLCM, local binary pattern (LBP), histogram of oriented gradient (HOG), and speeded up robust features (SURF) for extracting texture features. The look, structure, and arrangement of an object's pixels within a picture is referred to as texture. Next, we explain these feature extraction methods in more detailed.

1) GLCM

GLCM [17] is a statistical method that characterizes an image's texture by calculating how often a pair of pixels with specific values and specific distance occurs with exact specific length distance happen in an image. Mathematically, GLCM can be described as (1).

$$p_{(d,l)}(x,y) = \frac{C_{(d,l)}(x,y)}{\sum_{x,y} C_{(d,l)}(x,y)} \quad (1)$$

P (x, y) is the probability of two pixels with grey levels *x* and *y* located at distance l and direction *d*. And *C (x, y)* is a co-occurrence matrix that shows how often pairs of pixels with x and y values separated by l appear.

2) LBP

LBP is another technique for texture feature extraction. It is a gray scale-invariant and was first introduced by Ojala *et al.* [18]. The LBP is a texture description operator that computes the differences between adjacent and centre pixels. As shown in [13], the basic equation of LBP can be written as follows.

$$LBP(i_c, j_c) = \sum_{n=0}^{7} 2^n\, g\big(I_n - I(i_c, j_c)\big) \quad (2)$$

$LBP(i_c, j_c)$ is a LBP value at centre pixel (i_c, j_c). Index $I_n$and $I(i_c, j_c)$ are the values of the neighbour pixel and the centre pixel respectively. The index of neighbouring pixels is n. The image binary pattern is represented by the function *g(x)*.

3) HOG

HOG [19] is used for object detection. HOG descriptors are local statistics of the orientation of image gradient. HOG feature extraction process is explained with following basic steps: i) gamma normalization, ii) gradient computation, and iii) descriptor blocks and normalization.

The input image is evaluated with power-law equalization for effects on performance. An image sample is divided into small spatial regions called “cells” and each cell has some pixel size. Histogram of gradient direction or edge orientations are compiled for each pixel within these cells. Gradient magnitude and orientation are represented in (3) and (4) as in [20].

$$G(\varphi, w) = \sqrt{G_x(\varphi, \omega)^2 + G_y(\varphi, \omega)^2} \quad (3)$$

$$\theta(\varphi, w) = tan^{-1} \frac{G_y(\varphi, \omega)}{G_x(\varphi, \omega)} \quad (4)$$

Where G_x,G_y are derivatives with respect to *x* and *y* of an image I and the derivatives are computed in (5) and (6) with pixel differences.

$$G_x(\varphi, \omega) = I(\varphi + 1, \omega) - I(\varphi - 1, \omega) \quad (5)$$

$$G_y(\varphi, \omega) = I(\varphi, \omega + 1) - I(\varphi, \omega - 1) \quad (6)$$

Then each pixel within a cell plays a weighted vote for the local histogram of gradient directions. This histogram divides the gradient angle range into K bins. Then, normalize all cells within the block before using them for better invariance to illumination, and shadowing. Finally, all of the blocks' histograms were concatenated to create a HOG description.

4) SURF

The SURF is a novel scale and rotation invariant point detector and descriptor [21]. The SURF features extraction algorithm has three main steps: i) Selection of interest point; ii) Description of the neighbourhood of every interest point by a feature vector; and iii) matching of descriptor vector. Thus, based on this advantage, SURF features are extracted from images through the SURF detector and descriptor. The First 64 interest points per image are removed and then computed the described embodiment. Hessian matrix used in SURF:

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \quad (7)$$

The convolutions of the Gaussian second order partial derivatives with the image I in point x is $L_{xx}(x,\sigma), L_{xy}(x,\sigma), L_{xy}(x,\sigma), L_{yy}(x,\sigma)$ respectively. Hessian determinant using the approximated Gaussians:

$$det(H_{approx}) = D_{xx}D_{yy} - (\omega D_{xy})^2 \quad (8)$$

Where w is a weight for energy conservation between Gaussian kernel and approximated Gaussian kernel with ω=0.9.

2.2.5. Classification

Classification is the final process of identifying an object. Three key processes in constructing a precise and robust classifier for the recognition task are training, testing, and validation. The classifier is trained using the feature set received by feature extractors. Three different classifiers have been used in this study i.e., SVM, k-NN, and decision tree (DT) classifier, to classify tomato pest dataset.

1) SVM

The SVM [22] is a supervised machine learning technique based on statistical theory for pattern classification. The basic idea behind SVM is to use a hyperplane to separate the input to a decision performance, which is defined by $\vec{w}$ and bias b [23]. The (9) represents equation for hyperplane. Then the corresponding decision function yields as in (9). The maximum margin separated between two classes can be obtained by (10), and the hyperplane margin determines the sign $f(\vec{x})$. After solving the optimal hyperplane, the resultant is given in (11) and (12), where x_a and x_b are supporting vectors of two classes. Then decision function is given in (13).

$$\langle \vec{w}, \vec{x} \rangle b = 0, \text{ where} \vec{w} \in R^m, b \in R \quad (9)$$

$$f(\vec{x}) = sgn(\langle \vec{w}, \vec{x} \rangle + b) \quad (10)$$

$$w(\vec{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i\, y_j k(x_i x_j) \text{ where } i, j \in \{1,2.\ldots.n\} \quad (11)$$

$$w^* = \sum_{i=1}^{n} \alpha_i y_i \vec{x}_i \quad (12)$$

$$b^* = \frac{-1}{2}\langle \vec{w}^*, \vec{x}_a + \vec{x}_b \rangle \quad (13)$$

SVM is chosen because of its efficient implementations and its capability to work for great high dimensional problems with small datasets.

2) k-NN

The k-NN algorithm is proposed by fix and hodges, a popular supervised classification algorithm. The algorithm sorts fresh input data into k nearest neighbours groups, where k is a number that the user specifies. Furthermore, the label is allocated to the class that contains the bulk of the k-points. It has some limitations to computationally expensive and high memory. In contrast, it is a simple algorithm easy to understand. It uses similar data points to calculate the distance [24].

3) DT

The DT algorithm is proposed by Quinlan [25], the most widely used method for inductive inference. It has the advantage of self-explanatory logic flow classifiers, which handles the dataset with an error. The basic structure of DT is tree-like, with nodes representing features, branches representing experiments, and leaf nodes representing class labels. A decision criterion for such characteristic may be found in the root and every inside node. This classifier encounters a sample set split into two or more subsets. This technique is repeated until the defined standard has been met satisfactorily.

3. EXPERIMENTAL SET UP

We build a tomato pest dataset. In our dataset there are 859 number of images of 10 pest types. The size of all images is 256×256 with joint photographic experts group (JPEG) format. The experiment procedure was accomplished in MATLAB environment and open-source framework waikato environment

for knowledge analysis (WEKA). First, we loaded our dataset in MATLAB environment, then carrying out operations like pre-processing, segmentation and feature extraction in the same environment. Further, classification of extracted features using ML algorithms like SVM, k-NN, and DT was conducted in WEKA platform by setting some parameters. For all our experiments, simulations are done on a laptop with Intel® Core™2 DUO CPU T5750 @2GHz, 4GB RAM, and windows 7 operating system.

4. RESULTS AND DISCUSSION

The classification of tomato pests using the ML algorithm has been analyzed. In the above classification, we have used 10 GLCM features, 3776 LBP features,1764 HOG features, and 640 SURF features which were manually designed. The performances of three classifiers (SVM, k-NN, and DT) were evaluated with four features. The result is shown in Table 2. In addition, the lowest accuracy of 34.80% has been achieved by the SVM classifier with GLCM features. The comparison of all three aforementioned classifiers' (SVM, K-NN, and decision trees) performance in terms of accuracy in percentage (%) with different features are summarized in graphical form and represented as in Figure 3.

Table 2. Performance of accuracy in percentage (%) of different classifiers with different features

Classifier	GLCM	LBP	HOG	SURF
SVM	34.80	81.02	77.99	72.4
k-NN	78.23	78.23	75.43	73.22
DT	76.60	66.12	66.93	64.84

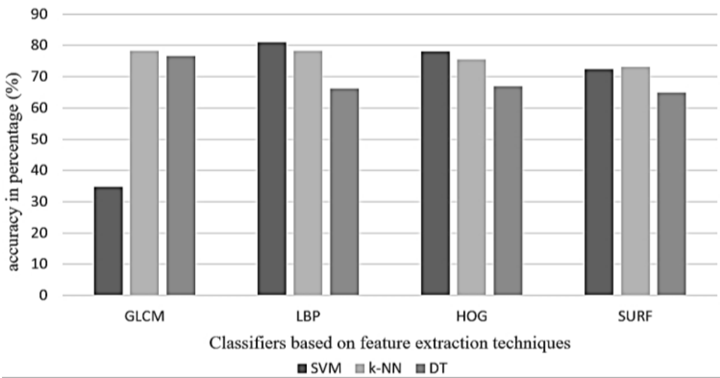

Figure 3. Comparison of different feature extraction techniques

From the Table 2 and Figure 3, it is observed that the best accuracy of 81.02% is obtained by using an SVM classifier with LBP features. Moreover, to validate our results, we have also calculated other evaluating parameters like sensitivity, specificity, precision, F-Measure and receiver operating characteristics (ROC) area for each class using highest accuracy obtained SVM classifier, which has shown in Table 3. Along with this, the ROC curve has also been investigated in Table 3 to illustrate the performance of the SVM classifier. The ROC curve compares true positive rates (TPR) to false positive rates (FPR) for various classification threshold levels. It is shown from Table 3 that pest 2 (Bemisia tabacii) has the highest ROC value of about 0.962.

Table. 2 Other performance parameter obtained using an SVM classifier with LBP based texture feature

Class	Sensitivity	Specificity	Precision	F-measure	ROC area
Pest 1	81.9	2.3	76.6	0.792	0.938
Pest 2	78.3	1.4	81.0	0.797	0.962
Pest 3	89.0	3.1	77.1	0.827	0.954
Pest 4	88.6	2.2	87.2	0.879	0.958
Pest 5	88.1	2.4	83.2	0.856	0.944
Pest 6	88.4	2.1	82.6	0.835	0.915
Pest 7	77.4	1.9	81.3	0.793	0.902
Pest 8	60.8	2.4	71.6	0.658	0.864
Pest 9	79.4	1.3	84.4	0.818	0.929
Pest 10	79.7	2.1	81.0	0.777	0.899

5. CONCLUSION

In this paper, we show how to distinguish tomato pest from an image using three distinct classification algorithms (SVM, k-NN, and DT) based on texture feature analysis (GLCM, LBP, HOG, and SURF). SVM classifier with LBP features had the maximum recognition accuracy of 81.02%. This research examines the use of machine learning in the early detection of tomato pest infestations, with the goal of improving tomato crop quality and productivity. Besides the maximum benefits of these machine learning approaches, there are certain limitations as it requires complex feature engineering work. To avoid these technique's drawbacks, combining different feature extraction method is a good idea for better accuracy.

REFERENCES

[1] G. S. Dhaliwal, V. Jindal, and B. Mohindru, "Crop Losses due to insect pests: Global and Indian Scenario," *Indian Journal of Entomology*, vol. 77, no. 2, pp. 165, 2015, doi: 10.5958/0974-8172.2015.00033.4.
[2] J. Wang, C. Lin, L. Ji, and A. Liang, "A new automatic identification system of insect images at the order level," *Knowledge-Based Systems*, vol. 33, pp. 102-110, Sep. 2012, doi: 10.1016/j.knosys.2012.03.014.
[3] F. Fina, P. Birch, R. Young, J. Obu, B. Faithpraise, and C. Chatwin, "Automatic Plant Pest Detection & Recognition Using K-Means Clustering Algorithm & Correspondence Filters," *International Journal of Advanced Biotechnology and Research*, vol. 4, no. 2, pp. 1052-1062, 2013. [Online]. Available: https://www.researchgate.net/publication/255483247_Automatic_plant_pest_detection_recognition_using_k-means_clustering_algorithm_correspondence_filters
[4] C. Xie *et al.*, "Automatic classification for field crop insects via multiple-task sparse representation and multiple-kernel learning," *Computers and Electronics in Agriculture*, vol. 119, pp. 123-132, 2015, doi: 10.1016/j.compag.2015.10.015.
[5] Q. He, B. Ma, D. Qu, Q. Zhang, X. Hou, and J. Zhao, "Cotton pests and diseases detection based on image processing," *TELKOMNIKA Telecommunication, Computing, Electronics and Control*, vol. 11, no. 6, pp. 3445-3450, 2013, doi: 10.11591/telkomnika.v11i6.2721.
[6] A. Dey, D. Bhoumik, and K. N. Dey, "Automatic Detection of Whitefly Pest using Statistical Feature Extraction and Image Classification Methods," *International Research Journal of Engineering and Technology*, vol. 3, no. 9, pp. 950-959, 2016. [Online]. Available: https://www.irjet.net/archives/V3/i9/IRJET-V3I9171.pdf
[7] M. A. Ebrahimi, M. H. Khoshtaghaza, S. Minaei, and B. Jamshidi, "Vision-based pest detection based on SVM classification method," *Computers and Electronics in Agriculture*, vol. 137, pp. 52-58, May 2017, doi: 10.1016/j.compag.2017.03.016.
[8] D. Xiao, J. Feng, T. Lin, C. Pang, and Y. Ye, "Classification and recognition scheme for vegetable pests based on the BOF-SVM model," *International Journal of Agricultural and Biological Engineering*, vol. 11, no. 3, pp. 190-196, 2018. [Online]. Available: https://www.ijabe.org/index.php/ijabe/article/view/3477/pdf
[9] R. Wang *et al.*, "A Crop Pests Image Classification Algorithm Based on Deep Convolutional Neural Network," *TELKOMNIKA Telecommunication, Computing, Electronics and Control*, vol. 15, no. 3, pp-1239-1246, 2017, doi: 10.12928/telkomnika.v15i3.5382.
[10] F. G. Zalom, "Pests,Endangered Pesticides and Processing Tomatoes," *ISHS Acta Horticulturae 613: VIII International Symposium on the Processing Tomato*, 2003, vol. 1, no. 71. doi: 10.17660/ActaHortic.2003.613.35.
[11] G. P. Prathibha, T. G. Goutham, M. V Tejaswini, P. R. Rajas, and K. Balasubramani, "Early pest detection in tomato plantation using image processing," *International Journal of Computer Applications*, vol. 96, no. 12, p. 22, 2014, doi: 10.5120/16847-6707.
[12] P. R. Shashank, S. S. Suroshe, P. K. Singh, K. Chandrashekar, S. M. Nebapure, and N. M. Meshram, "Report of invasive tomato leaf miner, Tuta absoluta (Lepidoptera: Gelechiidae) from northern India," *Indian Journal of Agricultural Sciences*, vol. 86, no. 12, pp. 1635–1636, 2016. [Online]. Available: https://www.researchgate.net/publication/311694275_Report_of_invasive_tomato_leaf_miner_Tuta_absoluta_Lepidoptera_Gelechiidae_from_Northern_India
[13] A. Gutierrez, A. Ansuategi, L. Susperregi, C. Tubio, I. Rankić, and L. Lenža, "A Benchmarking of Learning Strategies for Pest Detection and Identification on Tomato Plants for Autonomous Scouting Robots Using Internal Databases," *Journal of Sensors*, vol. 2019, p-5219471, 2019, doi: 10.1155/2019/5219471.
[14] N. A. Semary, A. Tharwat, E. Elhariri, and A. Hassanien, "Fruit-based tomato grading system using features fusion and support vector machine," in *Intelligent Systems' 2014*, pp. 401-410, 2015, doi:10.1007/978-3-319-11310-4_35.
[15] S. R. Rupanagudi, Ranjani B. S., P. Nagaraj, V. G. Bhat, and Thippeswamy G, "A novel cloud computing based smart farming system for early detection of borer insects in tomatoes," *2015 International Conference on Communication, Information & Computing Technology (ICCICT)*, 2015, pp. 1-6, doi: 10.1109/ICCICT.2015.7045722.
[16] U. Mokhtar, M. A. S. Ali, A. E. Hassanien, and H. Hefny, "Identifying two of tomatoes leaf viruses using support vector machine," *Information Systems Design and Intelligent Applications, Springer*, pp. 771-782, 2015, doi: 10.1007/978-81-322-2250-7_77.
[17] S. Murakami, K. Homma, and T. Koike, "Detection of small pests on vegetable leaves using GLCM," in *2005 ASAE Annual Meeting, American Society of Agricultural and Biological Engineers*, 2005, vol. 0300, no. 05, 2013, doi: 10.13031/2013.19109.
[18] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971-987, July 2002, doi: 10.1109/TPAMI.2002.1017623.
[19] E. Prakasa, "Texture Feature Extraction by Using Local Binary Pattern," *INKOM Journal*, vol. 9, no. 2, p. 45, 2016, doi: 10.14203/J.INKOM.420.
[20] X.-Y. Xiao, R. Hu, S.-W. Zhang, and X.-F. Wang, "HOG-based approach for leaf classification," in *International Conference on Intelligent Computing*, pp. 149-155, 2010, doi: 10.1007/978-3-642-14932-0_19.
[21] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF : Speeded Up Robust Features," in *European Conference on Computer Vision*, pp. 404-417, 2006, doi: 10.1007/11744023_32.
[22] M. W. Ashour, M. F. Hussin, and K. M. Mahar, "Supervised texture classification using several features extraction techniques based on ANN and SVM," *2008 IEEE/ACS International Conference on Computer Systems and Applications*, 2008, pp. 567-574, doi: 10.1109/AICCSA.2008.4493588.
[23] T. Rumpf, A.-K. Mahlein, U. Steiner, E.-C. Oerke, H.-W. Dehne, and L. Plümer, "Early detection and classification of plant diseases with support vector machines based on hyperspectral reflectance," *Computers and Electronics in Agriculture*, vol. 74, no. 1, pp. 91–99, 2010, doi: 10.1016/J.COMPAG.2010.06.009.

[24] D. C. Corrales, J. C. Corrales, and A. Figueroa-Casas, "Towards detecting crop diseases and pest by supervised learning," *Ingeniería y Universidad*, vol. 19, no. 1, pp. 207-228, 2015. [Online]. Available: https://pdfs.semanticscholar.org/a4f7/c243e485e3b01626e4315db0e8723d70d7b5.pdf?_ga=2.168941985.1831004169.1643870076-894752331.1639972456
[25] J. R. Quinlan, C4. 5: Programs for machine learning, *Morgan Kaufmann Publisher Inc.*, San Francisco, pp 235-240, 1994. [Online]. Available: https://www.elsevier.com/books/c45/quinlan/978-0-08-050058-4

BIOGRAPHIES OF AUTHORS

Mrs. Gayatri Pattnaik has 10 years of teaching and research experience. She did her B.E. and M. Tech from Biju Patnaik University of Technology, Rourkela, Odisha and pursuing Ph.D. at Kalinga Institute of Industrial Technology. She has added one SCIE, one SCOPUS indexed journal, one book chapter and two international conferences to her credibility. Currently, she has submitted her thesis. She can be contacted at email: gayatripattnaik13@gmail.com.

Dr. Kodimala Parvathi is a teacher and administrator with 25 years of experience in teaching, research, and administration. She earned her bachelor's degree from Osmania University in Hyderabad and her master's and doctoral degrees in engineering from Andhra University in Visakhapatnam. She has papers in 15 international journals and 10 international conferences to her credit. She also generated three PhD scholars, and four more are still working under her supervision. She is a life member of various professional bodies like IET, ISTE. She can be contacted at email: kparvati16@gmail.com.